Phân Tích Dữ Liệu Thực Tế với Python

Bài 4.2: Các Loại Biểu Đồ Cơ Bản với Matplotlib

Quang-Khai Tran, Ph.D
CyberLab, 02/2023

(Ảnh: Internet)

CYBERLEARN
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

CYBERSOFT
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

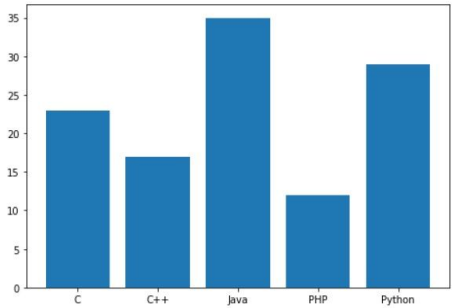
CYBERLAB
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

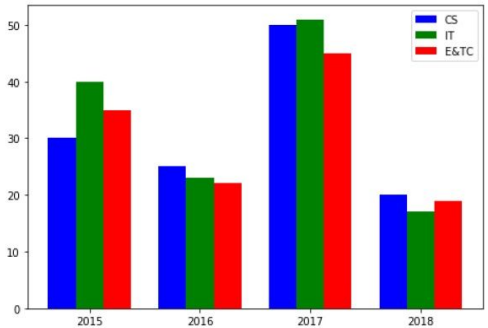
Nội dung

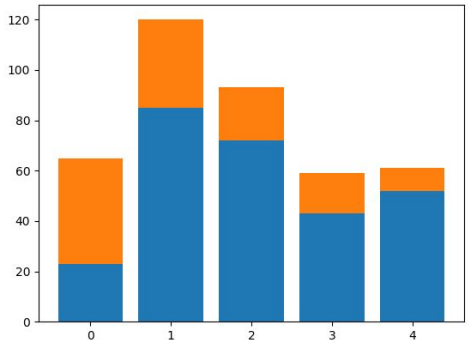
Phần 1. Các loại biểu đồ cơ bản

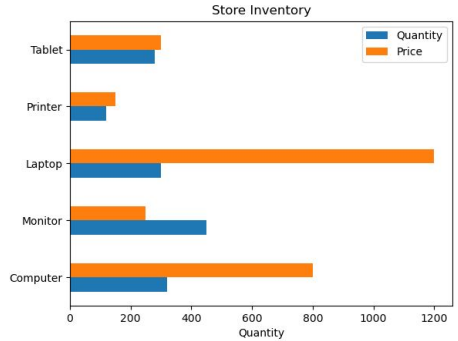
1.1 Biểu đồ cột

- Biểu đồ cột dùng để biểu diễn dữ liệu theo loại cùng giá trị của mỗi loại
- Một số loại biểu đồ cột:
  - Biểu đồ cột đơn
  - Biểu đồ cột nhóm
  - Biểu đồ cột chồng lên nhau
  - Biểu đồ cột dạng nằm ngang







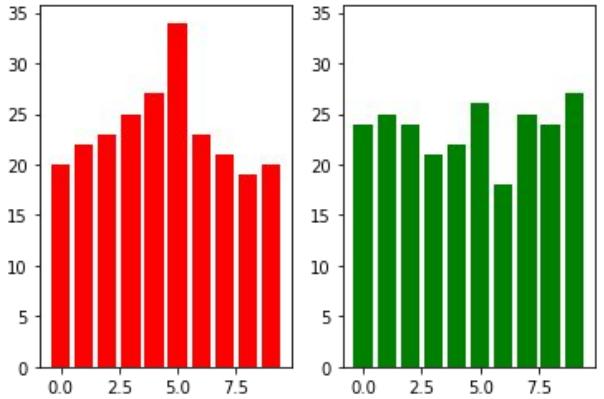


1.1 Biểu đồ cột

Hàm: `plt.bar(x, height, width, label, color, bottom,...)`

`plt.bar(x=[các vị trí trên trục x], height=[dữ liệu])`

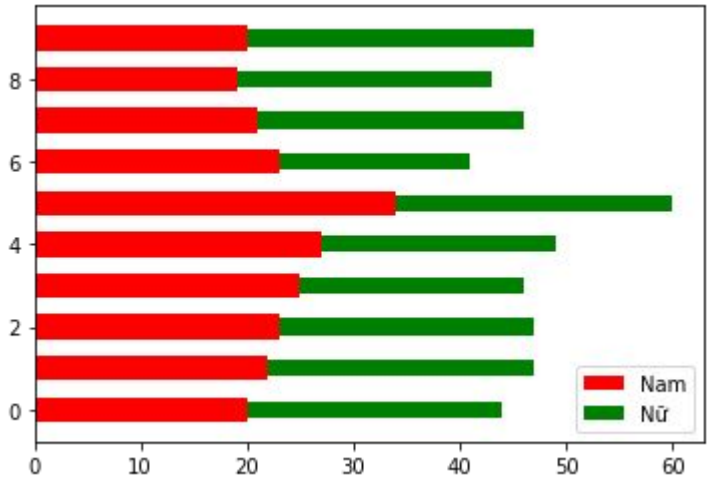
```
fig = plt.figure()
ax1 = fig.add_subplot(1, 2, 1)
ax2 = fig.add_subplot(1, 2, 2, sharey=ax1)

ax1.bar(x=range(len(ds1)),height=ds1, color="red")
ax2.bar(x=range(len(ds2)),height=ds2, color="green")

plt.show()
```



1.1 Biểu đồ cột

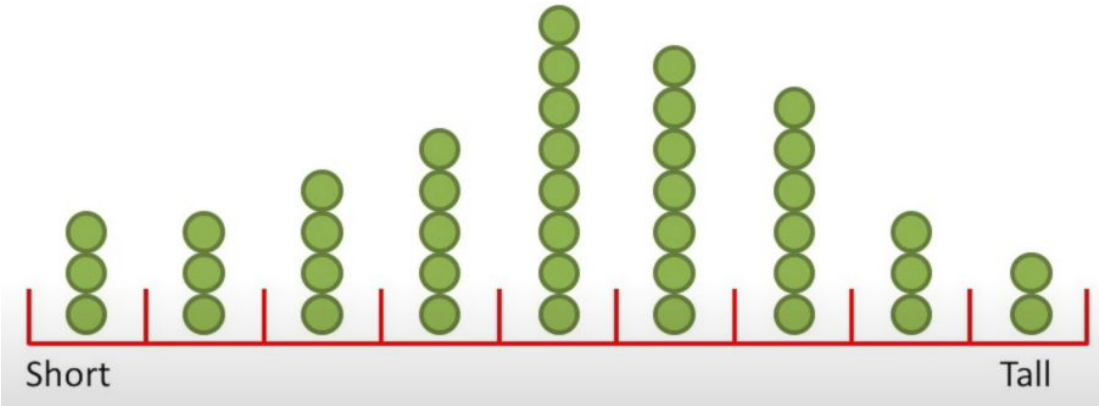
Vẽ biểu đồ theo chiều ngang (cùng stacked bar graph):

```
plt.barh(x, width, height, label, color, left,...)
```



1.2 Biểu đồ histogram

- Biểu đồ histogram là một biểu diễn “chính xác” về phân bố của dữ liệu
- Để vẽ biểu đồ histogram, cần xác định:
  - Bins: là range (khoảng) dữ liệu
  - Intervals: chia bins ra thành số các cột hiển thị dữ liệu
  - Đếm xem trong mỗi cột có bao nhiêu dữ liệu rơi vào



1.2 Biểu đồ histogram

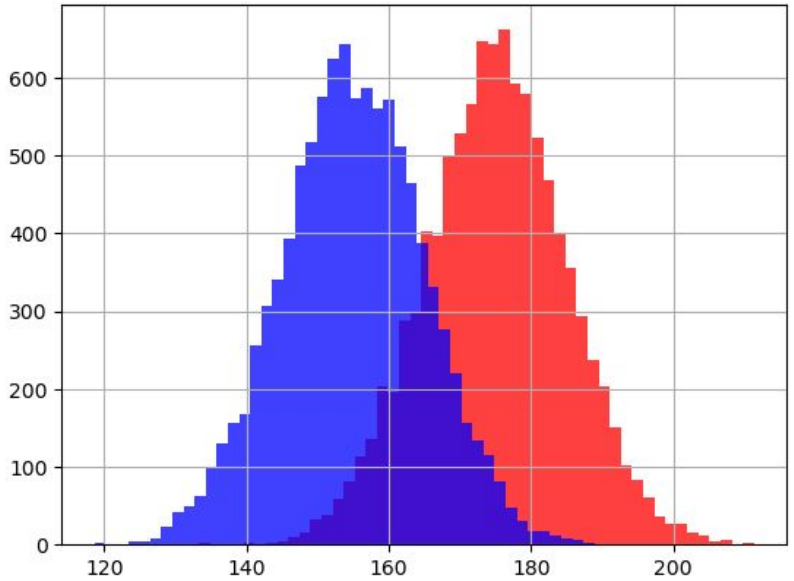
❖ Hàm vẽ histogram

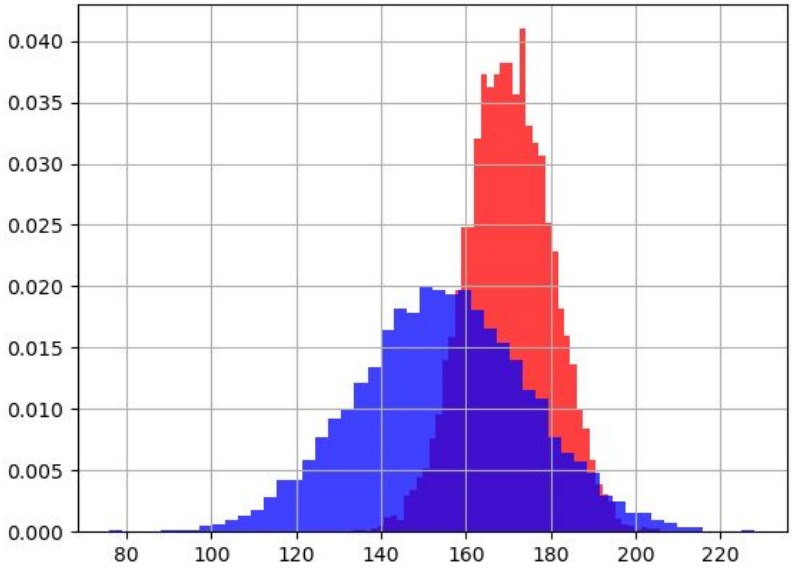
```
plt/ax.hist(x, bins[, range, density, cumulative, colors,
orientation, width, rwidth] )
```

x	mảng hoặc chuỗi các mảng (chứa dữ liệu cùng loại)
bins	integer hoặc list
range	tuple (min, max) xác định xác định khoảng của bins
density (True/False)	Nếu là True: hiển thị xác suất thay vì tần số
cumulative (True/False)	Nếu là True: ở mỗi bin sẽ là tổng của nó với tất cả các bins trước đó
color/facecolor	màu hoặc chuỗi các màu
orientation	hướng (‘vertical', ‘horizontal')
width, rwidth	độ rộng của cột, độ rộng tương đối so với kích thước bin

1.2 Biểu đồ histogram

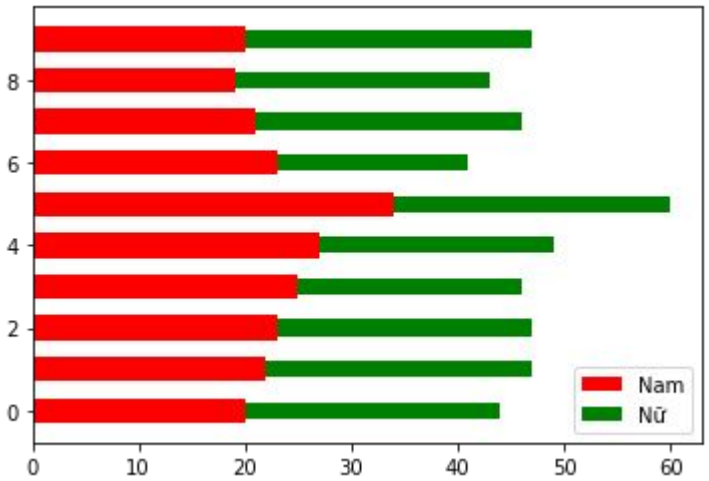
- So sánh phân bố của 2 thông số dữ liệu
  ⇒ Vẽ phân bố của 2 thông số trên cùng một histogram

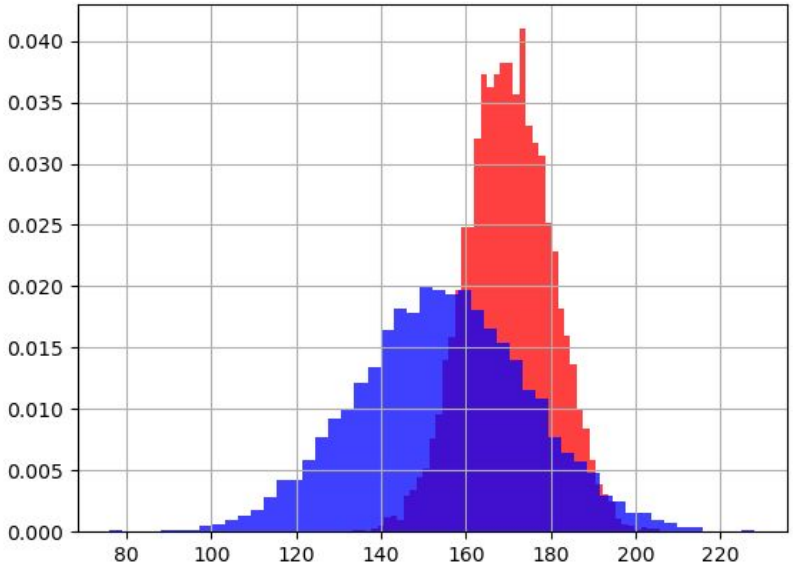




1.2 Biểu đồ histogram

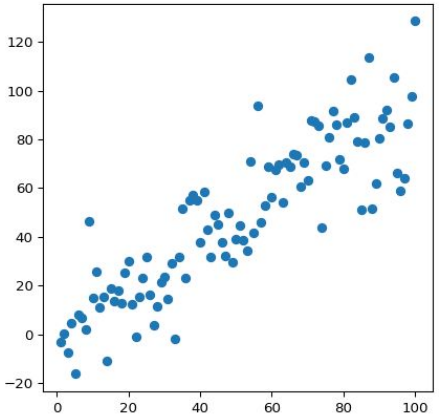
- Biểu đồ cột vs. histogram?
  - Cột: biểu diễn số lượng của dữ liệu (nhiều categories)
  - Histogram: biểu diễn phân bố của dữ liệu (1 category)

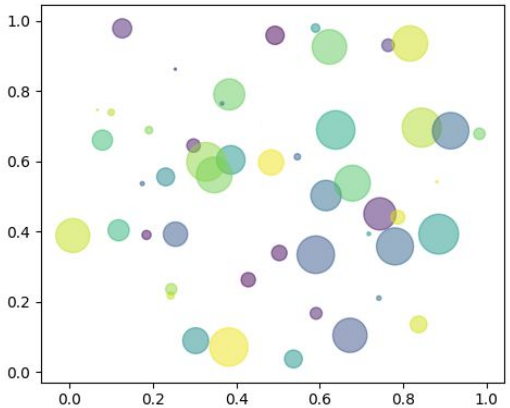


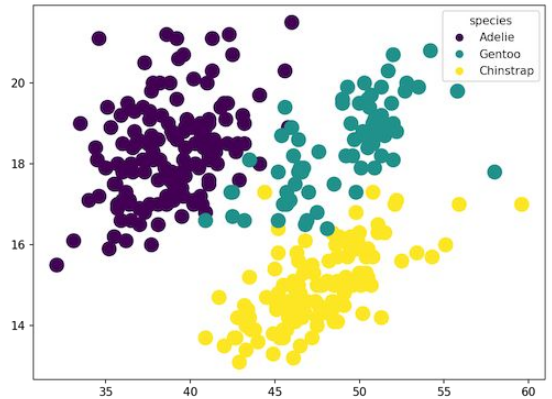


1.3 Biểu đồ scatter

- Dùng để biểu diễn dữ liệu theo 2 trục tung-hoành
- Nhằm xác định/chỉ ra mỗi biến ảnh hưởng/liên quan đến biến kia như thế nào
- Mỗi điểm dữ liệu là một dấu marker trên biểu đồ
- Có thể thêm chiều thông tin bằng cách sử dụng
  - màu sắc
  - kích thước







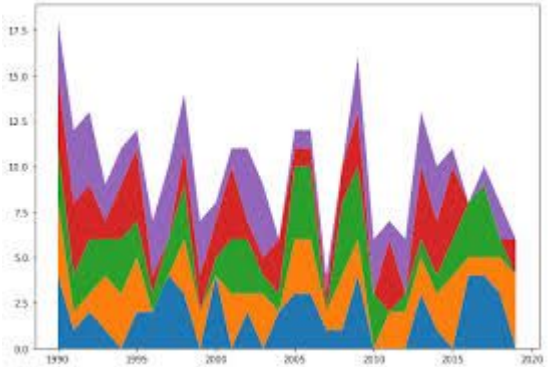
1.3 Biểu đồ scatter

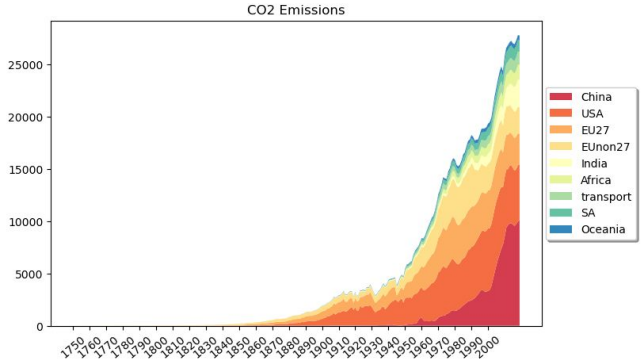
```
plt/ax.scatter(x, y[, s=None, c=None, marker=None, alpha=None,
linewidths=None, edgecolors=None, cmap=None, norm=None,
vmin=None, vmax=None])
```

x, y	số thực hoặc mảng số thực
s	số thực hoặc mảng số thực để xác định size của điểm
c	list các màu muốn tô
marker	kiểu của marker
alpha	độ mờ/rõ của màu tô
linewidths edgecolors	kích cỡ và màu của đường viền
cmap, norm, vmin, vmax	Các tham số xác định dải màu

1.4 Biểu đồ area

- Tương tự như biểu đồ dạng line nhưng chồng lên nhau
- Khoảng trống giữa line ở dưới cùng và trục x được tô màu
- Khoảng trống giữa line trên và line dưới cũng được tô màu
- Có 2 cách để vẽ biểu đồ dạng area
  - `plt.stackplot()`
  - `plt.fill_between()`





1.4 Biểu đồ area

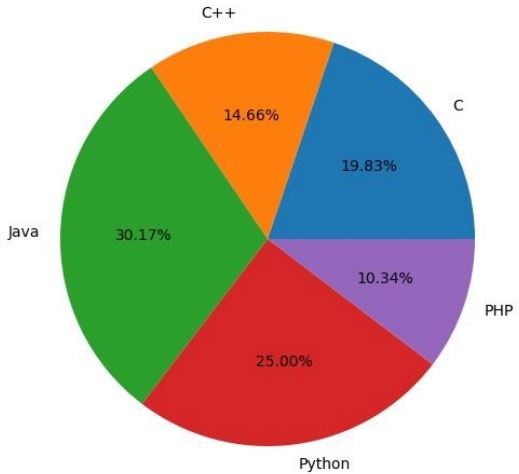
❖ Hàm `plt.stackplot()`

```
stackplot(x, *args, labels=(), colors=None,
baseline='zero', /                  data=None, **kwargs)
```

*args	(list của) các list chứa giá trị của từng thông số
labels	list các nhãn của từng thông số
colors	list các màu muốn tô
baseline	xác định cách tính baseline {'zero', 'sym', 'wiggle', 'weighted_wiggle'}

1.5 Biểu đồ hình tròn

- Biểu đồ hình tròn (pie) thể hiện tỷ lệ phần trăm của một chuỗi dữ liệu
- Chỉ thực hiện với một chuỗi dữ liệu



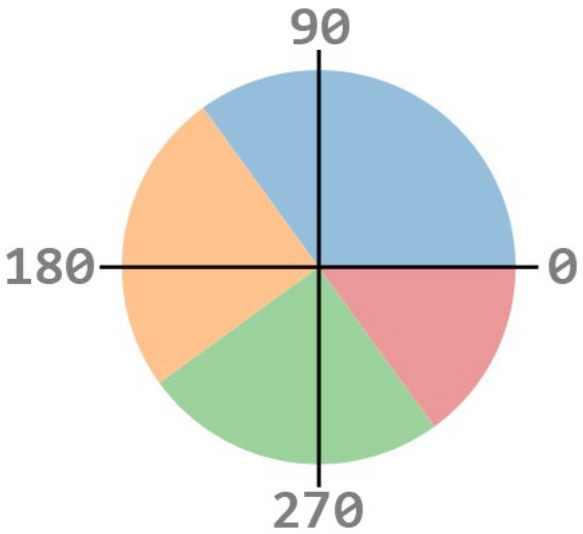
1.5 Biểu đồ hình tròn

Hàm: `plt.pie(sizes[, explode, labels, autopct, colors, shadow=True, startangle=0])`

sizes	list hoặc mảng (chứa dữ liệu cùng loại)
explode	list hoặc set xác định một số vị trí được tách ra
labels	nhãn của các thông số dữ liệu
autopct	format của nhãn phần trăm
colors	list các màu
shadow	đổ bóng
startangle	góc bắt đầu

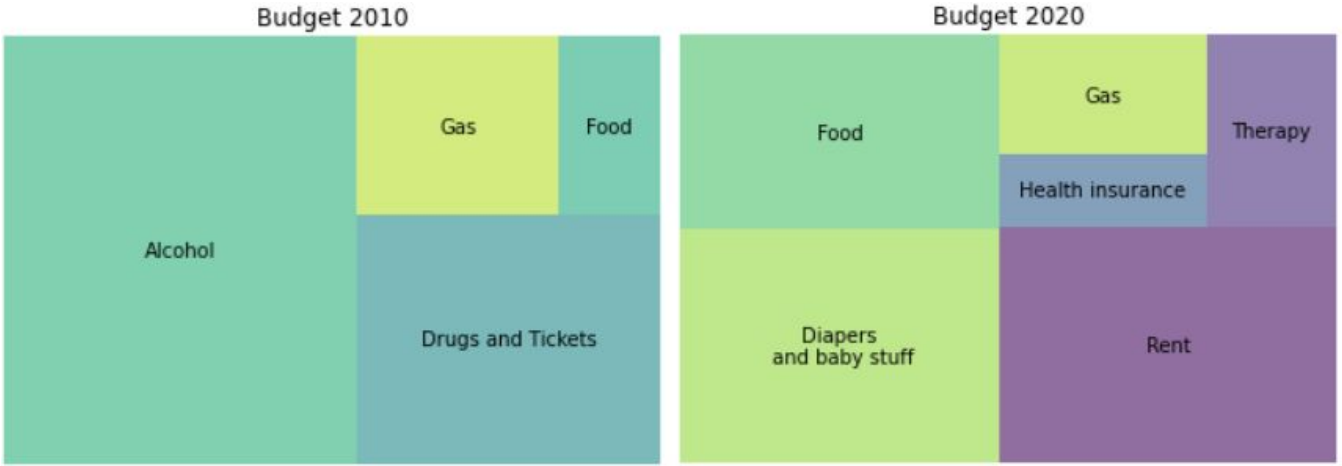
1.5 Biểu đồ hình tròn

Hàm: `plt.pie(sizes, explode, labels, autopct, colors, shadow=True, startangle=0)`



1.6 Biểu đồ Tree-Map

- Treemap hiển thị dữ liệu phân cấp
  - Gồm các hình chữ nhật lồng ghép
  - Thể hiện được tương quan về lượng giữa các thông số



1.6 Biểu đồ Tree-Map

- Được phát minh bởi Ben Shneiderman (1990), giáo sư Computer Science tại trường University of Maryland

1.6 Biểu đồ Tree-Map

- Có nhiều module/thư viện hỗ trợ vẽ Treemap
- Demo: Sử dụng module `squarify`
- Cần cài đặt vào Anaconda:
  1. Mở terminal với Anaconda (trong Windows, dùng Command Prompt)
  2. Chạy dòng lệnh: conda install squarify
     (hoặc conda install -c conda-forge squarify)

- Hoặc chạy lệnh %pip install squarify trên Notebook

```
import squarify
squarify.plot(sizes=volume, label=labels,
              color=color_list, alpha=0.7)
plt.axis("off")
plt.show()
```

2 Bài tập

1. Sử dụng dữ liệu bán hàng 'sale_data_vn.csv':
   - Vẽ biểu đồ cột, pie về tần số, tuần suất đơn hàng theo
     - Địa phương (HN, SG, ĐN)
     - Giới tính
   - Vẽ biểu đồ histogram về phân bố giá trị đơn hàng, rating
     - Phân bố chung
     - Theo giới tính, theo địa phương
   - Vẽ biểu đồ scatter về mối liên hệ giữa giá trị đơn hàng và rating
   - Vẽ biểu đồ tree-map cho các mặt hàng ('Product line')
2. Rút ra nhận xét cho các biểu đồ ở trên
3. Yêu cầu:
   - Bố cục trình bày: cần tổ chức các subplots phù hợp cho từng yêu cầu trên

THANK YOU!